REMARKS

This Amendment is submitted in response to the Examiner's Action mailed March 29, 2004, with a shortened statutory period of three months set to expire June 29, 2004. With this amendment, claims 1-3, 5, 9, 20, 30-31, 34, and 38 have been amended; claims 10-19, 25-29, and 39-50 have been canceled; and claims 51-59 have been added.

The Examiner rejected claims 49-50 under 35 U.S.C. § 101 stating that the claimed invention lacks patentable utility. The Examiner stated that these claims failed to provide a tangible process, machine, manufacture, or composition of matter and was merely an arrangement of data bits. These claims have been canceled. Therefore, this rejection should be withdrawn.

The Examiner rejected claim 11 under 35 U.S.C. § 112, first paragraph, as failing to comply with the enablement requirement. The Examiner also rejected claim 11 under 35 U.S.C. § 112, second paragraph, as being indefinite stating that there is insufficient antecedent basis for the term "the interrupt". This claim has been canceled. Therefore, these rejections should be withdrawn.

The Examiner rejected claims 1, 8, 10, 12, 13, 20-22, 25-27, 30, 37, 39, 41, 42, 49, and 50 under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent 6,625,749 issued to *Quach*. This rejection, as it might be applied to the claims as amended, is respectfully traversed.

Applicants have amended the claims to describe a method, system, and product for preventing a data processing system from crashing when a parity error occurs in a cache that is associated with a processor in the system. The parity error is processed during runtime of the system. Responsive to an occurrence of a parity error in the cache that is associated with the processor, a single set of processor information is stored in registers in the processor to a form stored processor information. The processor information indicates an error type, error states, and error status that is stored as a result of the occurrence of the parity error. A determination is made as to whether the parity error is a recoverable error using the stored processor information by reading the registers to analyze the stored error type, states, and status. Some examples of support for these

amendments can be found in the specification on page 10, lines 8-28, page 13, lines 3-17, and page 15, lines 22-31.

Quach teaches a processor that includes redundant processing cores. The processor is capable of operating in a redundant mode or a split mode. In redundant mode, the processing cores operate in lock step on identical code segments and compare the results to identify errors. See column 3, lines 34-36. In split mode, the cores can operate independently. Uncorrupted data may be stored to a designated memory location. A core can then be initialized with processor state data recovered from this saved data. See column 4, lines 9-28. *Quach* teaches recovering from a parity error by restoring saved uncorrupted data.

Quach does not teach storing a single set of processor information that indicates error states, type, and status in registers in the processor in response to the occurrence of a parity error. *Quach* does not teach storing error type, states, and status at all. *Quach* teaches merely how to use uncorrupted processor state information to recover from a parity error. *Quach* does not teach determining whether the parity error is a recoverable error by reading the registers to analyze the stored error type, states, and status.

The Examiner rejected claims 2-7, 14-19, 31-36, and 43-48 under 35 U.S.C. § 103(a) as being unpatentable over *Quach* in view of U.S. Patent 5,659,678 issued to *Aichelmann*. This rejection, as it might be applied to the claims as amended, is respectfully traversed.

The claims further describe the error type that is saved in processor registers in response to the occurrence of a parity error. The error type is saved in registers in a processor that is associated with a cache in which the parity error occurred. *Quach* provides no teaching about storing error type, states, and status in response to the occurrence of a parity error. *Quach* provides no motivation to combine *Quach* with any other reference with regard to the storage of error type, states, and status. *Quach* provides no motivation to combine *Quach* with any other reference with regard to the storage of error type, states, and status in registers in a processor that is associated with a cache in which a parity error occurred. *Quach* provides no teaching about determining whether a parity error is a recoverable error. Thus, *Quach* provides no motivation for combination with any other reference with regard to determining whether a parity error is

a recoverable error by reading processor registers to analyze stored error type, states, and status.

The Examiner states that *Quach* does not teach where an error count should be kept or that when the count exceeds a threshold that the memory cache should be disabled. The Examiner uses *Aichelmann* to supply these missing features stating that *Aichelmann* teaches an error count and disabling a memory cache when the count exceeds a threshold.

The combination of *Quach* and *Aichelmann* does not describe, teach, or suggest storing a single set of processor information in a processor's registers in response to an occurrence of a parity error where the information indicates an error type, states, and status and that are stored as a result of the occurrence of a parity error. The combination does not describe, teach, or suggest determining whether the error is a recoverable error by reading these registers to analyze the stored error type, states, and status. The combination does not describe, teach, or suggest if the error type that is indicated in the registers in the processor is one of a level one cache or a level one cache tag error, incrementing an error count, or if the error count is greater than a threshold, disabling the failing portion of the level one data cache or data cache tag.

The Examiner rejected claims 9, 11, 23, 28, 38, and 40 under 35 U.S.C. § 103(a) as being unpatentable over *Quach* in view of U.S. Patent 6,615,374 issued to *Moran*. This rejection, as it might be applied to the claims as amended, is respectfully traversed.

Claim 9 describes if the parity error is a non-recoverable parity error, generating an indication of the non-recoverable interrupt error and storing the indication in the processor's registers. The Examiner states that *Quach* does not teach indicating that the parity error is a non-recoverable type. The Examiner uses *Moran* to supply this missing feature.

Moran teaches storing an error condition in a system address chip 130, a system data chip 140, or a memory card MAC 156. *Moran* does not teach storing the error condition in registers in a processor. *Moran* does not teach storing the error condition in registers in a processor that is associated with a cache in which the error occurred. The combination of *Quach* and *Moran* does not describe, teach, or suggest if the parity error

is a non-recoverable parity error, generating an indication of the non-recoverable interrupt error and storing the indication in the processor's registers.

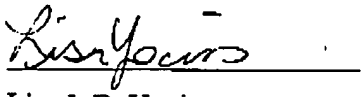
The Examiner rejected claims 24 and 29 under 35 U.S.C. § 103(a) as being unpatentable over *Quach* in view of U.S. Patent 6,445,717 issued to *Gibson*. This rejection, as it might be applied to the claims as amended, is respectfully traversed.

Claim 29 has been canceled. Claim 24 describes the communications unit being one of a modem and Ethernet adapter in combination with a processing unit that is connected to the bus system, where the processing unit includes registers for storing a single set of processor information that indicates an error type, error states, and error status to form stored processor information in response to the detection of a parity error in the cache that is associated with the processor and determining whether the parity error is a recoverable parity error using the stored processor information by reading said registers to analyze said stored error type, error states, and error status. The combination of *Quach* and *Gibson* does not describe, teach, or suggest the combination of these features.

Applicants' claims are believed to be patentably distinct over the prior art because the prior art does not describe, teach, or suggest Applicants' claims. The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: 06.29.04

Respectfully submitted,



Lisa L.B. Yociss
Reg. No. 36,975
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 367-2001
Attorney for Applicants